Peer-to-Peer Netw. Appl.
DOI 10.1007/s12083-016-0537-1

Multi-class routing protocol using virtualization and SDN-enabled architecture for smart grid

Mohammad Alishahi[1] · **Mohammad Hossein Yaghmaee Moghaddam**[2] · **Hamid Reza Pourreza**[3]

Received: 3 March 2016 / Accepted: 7 December 2016

Abstract Routing Protocol for Low Power and Lossy Network (RPL) is standardized and known as the primary solution for the last mile communication network in the smart grid. Various applications with different requirements are rapidly developed in the smart grid. The need to provide Quality of Service (QoS) for such a communication network is inevitable. In this paper, we use the benefits of virtualization and software-defined networking to present a virtual version of the RPL protocol which we name OMC-RPL (Optimized Multi-Class RPL). We present an SDN-enabled architecture consisting of a central controller and some SDN nodes. This implementation reduces the complexity and controls interactions to distribute the network states and other related information in the network. The proposed SDN-enabled architecture consists of different components including Network Link Discovery, Topology Manager, and Virtual Routing. OMC-RPL utilizes a holistic objective function including distinctive metrics related to QoS, and supports the data classification which is an essential requirement in this context. The proposed objective function considers different numbers of traffic classes by using weighting parameters. An optimization algorithm determines the best values of these coefficients. OMC-RPL is evaluated in different aspects. Simulation results show that the new idea significantly decreases both the end-to-end delay and packet loss which are the important factors of QoS. The virtualization idea is also investigated, which results in less message exchange.

Keyword Smart grid · RPL · QoS · SDN · Virtualization

✉ Mohammad Hossein Yaghmaee Moghaddam
yaghmaee@ieee.org

Mohammad Alishahi
m.alishahi@srbiau.ac.ir; alishahi.1983@gmail.com

Hamid Reza Pourreza
hpoureza@ieee.org; hpourreza@um.ac.ir

[1] Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran

[2] Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

[3] Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

1 Introduction

One of the primary issues in smart grid is establishing a reliable and secure communication network that can support different applications such as Advanced Metering Infrastructure (AMI), Demand Response (DR), Distribution Automation (DA), and so on. Within the communication network associated with the power grid, the Neighbor Area Network (NAN) and Home Area Network (HAN), are faced with substantial communication challenges because of their size and traffic variation [1–3]. The IETF ROLL working group had a mission to propose a routing protocol for LLN (Low power and Lossy Networks) which lead to the RPL (Routing Protocol for LLN) standard in 2012 [4, 5]. LLNs are made up of a large number of embedded devices with limited power, memory, and resources that connect to each other using various communications protocols, such as IEEE 802.15.4, Wi-Fi, and Power-Line Communication (PLC) [6]. RPL is the main candidate for acting as the standard routing protocol for IP smart object networks in NAN. This popularity is because of two reasons, one is its flexibility to adapt to different topologies, and the other is its capability of QoS support [7]. RPL is a distance-vector protocol that is based on the concept of a topological Directed Acyclic Graph (DAG). DAG uses a tree structure in which each node can have more than one parent. Specifically, RPL organizes these nodes into Destination Oriented DAGs (DODAG) whose roots are destination nodes—e.g., sinks, or network gateways. DODAGs are

Published online: 19 December 2016

created and managed based on the Objective Function (OF). The objective function specifies routing metrics and optimization goals. It can construct routes to satisfy any requirements, such as quality of service. To construct a DODAG, the root sends the Objective Code Point (OCP), which identifies the objective function and the other related information via a standard IPv6 message to its neighbor's nodes. The DODAG's creation is finalized when the nodes decide, using a general algorithm, their preferred parent, and rank. The rank of a node [8] is also computed by the objective function, which expresses the distance of the node from its root in relation to the given metrics; nodes closer to the root should have lower ranks. The RPL protocol's process helps to create a self-configuring, self-healing, loop detecting system that will be suitable for NAN and HAN networks in the smart grid. The specification of RPL does not force any routing metric and left it open to implementations. The proposed objective functions by the IETF presented in [8] and [9] have defined some recommendations on how to implement objective function without specifying available routing metrics. In RFC 6552 [8] the principle of the objective function is described, which is called Objective Function Zero (OF0). As explained earlier, the two most important duties of an objective function are choosing a proper rank and the preferred parents. RFC 6552 describes the principles and rules of defining an objective function based on the required metrics and constraints. For instance, there is a rule that says a node with the lowest rank should be chosen as a preferred parent, but note that this document does not consider any routing metric specified in [10]. The proposed objective function in this paper is also based on these foundations. One typical objective function based on the metric of link quality is Expected Transmission Count (ETX) [9]. The main idea of this objective function is the probable amount of transmission to send a packet successfully. It is usually used in wireless environments. ETX has been widely used in the different studies [11, 12].

Distinctive types of applications in the smart grid, especially in NAN and HAN are experiencing the same situation as LLN. This last mile network is made of highly limited devices interconnected by fairly unstable low-quality links that cause the different quality of service requirements, which is not the same as the traditional IP networks [13]. Also, the QoS is an essential component of the overall architecture in the smart grid [14]. Some data, such as alerts or control signals, have real-time requirements. Therefore, the networking infrastructure should somehow guarantee the quality of service, for example, decrease the end-to-end delay.

Software Defined Networking (SDN) is a major trend in the telecommunication industry today. In the SDN, the control and data planes in each network equipment (NE) is separated and logically centralized. In this case, each NE just forwards the traffic and enforces policy according to instructions received from the controller. This makes the network programmable in a way that promises to be more flexible than the currently managed paradigm. In software-defined networks, the interface between applications and networks has been changed. This makes it more suitable for developing different applications in the smart grid which need a higher degree of network awareness. Ease of configuration and management, cross-domain content-based networking, and virtualization and isolation are some opportunities of SDN in smart grid networks [15]. Virtualization of Sensor Network (VSN), on the other hand, is a quite new research approach. A virtual sensor network is formed by providing logical connectivity among a subset of sensor nodes that are dedicated to a certain task or application at a given time [16]. Virtual sensors are software sensors that are built on top of actual physical sensors. They have some data processing functions for complex queries. These functions combine and process sensed data from a group of heterogeneous sensors [17]. Each virtual sensor is created from one or more physical sensors based on the task they perform. Users can freely create and use virtual sensors as if they owned them. These sensors are accessed on-demand, and they are deployed when they are an active request for using data [18]. The relationship between virtual sensors and physical sensors is illustrated in Fig. 1. VSN is a logical subset of a physical sensor network, which is required for serving the particular application with two or more nonadjacent sensor nodes. In fact, it abstracts the complexities of setting communication link between nodes. Sensor Cloud (SC), on the contrary, delivers Sensor-as-a-Service (Se-aaS) by virtualizing the physical resources [19].

In this paper, by utilizing the benefits of SDN and virtualization, we propose a virtualization based QoS-aware multi-class routing protocol for low-power and lossy network used in AMI networks in the smart grid. The main contributions of the current study are summarized as follows:

– We develop a new centralized architecture for multi-class QoS routing in the smart grid using virtualization and software defined networking

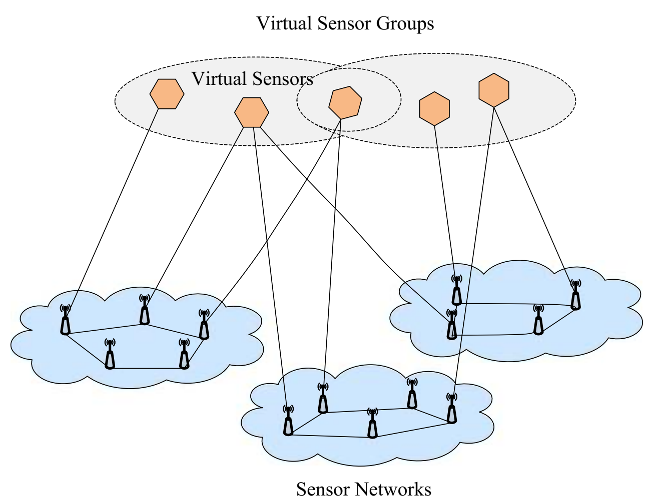


Fig. 1 The relation between virtual and physical sensors

– We propose a holistic objective function including distinctive metrics related to QoS
– We present OMC-RPL (Optimized Multi-Class RPL), a new extension of traditional RPL, which can support different classes of traffic.

The remainder of this paper is organized as follows. In section 2, some related work on SDN and QoS in the smart grid, especially those methods using RPL, are studied. Section 3 explains the proposed SDN-based architecture and the novel objective function. Finally, the last two sections include the simulation results and conclusion.

2 Related work

There are many studies on QoS in the smart grid. Some of them focus on challenges and requirements of QoS in this area. For instance, [20] discusses that one of the most important requirements is that each system architecture should support a diverse set of QoS classes with a wide range of rate and delay requirements. Other studies usually propose specific ways that somehow improve the QoS in the smart grid. For example, [21] presents a survey of QoS requirements for smart grid applications and then explain how to use of extended Differentiated Services Code Points (DSCPs) of Differentiated Service (DiffServ) approach to classifying smart grid traffics into different service classes. The authors in [22] study about the scheduling and routing methods based on Back-Pressure algorithm to guaranty the QoS in the smart grid network. They propose a multi-gate network structure and develop a packet-scheduling algorithm aimed at providing reliable two-way communications from meters to the AMI head-end. The trade-off between maximizing the throughput and minimizing the average overall network delay has been investigated.

In [23] first a review on different technologies for Neighborhood Area Networks (NAN) in the AMI networks is presented then a detailed description of the different routing protocols proposed for the NAN domain is introduced. A performance comparison of different routing protocols including RPL based on a set of metrics is presented.

In [24], two MAC-based routing metrics are used. The first one checks the ETX and the packet losses due to the MAC contention. The second metric selects the routes that have the acceptable traffic load by considering the power consumption and the application required reliability. The author implements the proposed method in a real testbed composed of seven Telosb motes. They evaluate the performance in terms of end-to-end reliability and the power consumption. In [25], the impact of objective functions on the network topology is analyzed. Link Quality Level (LQL) is another objective function, which is based on the link condition. The author compares two objective functions: OF0 and LQL. In [26] a combination of two routing metrics among hop counts, ETX, remaining energy and RSSI (Received Signal Strength Indication) are used. The first metric is responsible for choosing a parent with the lower rank. If the first values are equal, then the node with the lower rank of the second composition metric is selected as the preferred parent [13] suggests a cross-layer QoS mechanism that merges a priority queue with multiple instances of RPL. It focuses on the MAC level QoS separation. Moreover, the both RPL instances are based on the same objective function and root but generate distinct DODAGs due to the partitioning of the actual physical network. Nodes are classified as regular or alarm. The regular nodes are responsible for physical environment monitoring and generate data packets at a low rate; however, alarm nodes randomly generate small-size alert packets. The author intends to extend the idea of QoS through multiple RPL instances by supporting priority traffic in MAC layer and exploring the effect of traffic differentiation at the network layer. In [3], the QoS is guaranteed through traffic prioritization in MAC layer in a way that the random backoff mechanism is altered based on the traffic classes, and this is how they control the channel accessibility. The author compares the single instance RPL, multi-instance RPL and multi-instance RPL with prioritized channel back off to see the effect of traffic differentiation at the network layer [27] believes that in order to optimize the path to the DODAG's root, the existing objective functions rely either on a single metric or on the combination of two metrics. Thus, a novel objective function based on the fuzzy parameters has been designed. Four different metrics, including end-to-end delay, hop counts, link quality and remaining energy are used to propose holistic objective function by using fuzzy logic. The proposed fuzzy system is a four-input controller with three membership functions for each input that leads to 81 rules. Eventually, by using centroid defuzzification method, control action based on several membership values is produced. In [28] a fair comparison between RPL as a proactive candidate, and LOADng (LLN On-demand Ad-hoc Distance vector routing protocol—next generation) as a reactive candidate is presented. Different real traffic scenarios and topology deployments of varied size with random topologies have been utilized. The bounds on control overhead for the two protocols has been derived. Results indicate that RPL has lower overhead than LOADng. It has been confirmed that in general, RPL outperforms LOADng for several critical metrics. The authors in [29] propose some enhancements to the RPL to provide QoS guarantees for static and mobile LLNs. They propose a fuzzy logic objective function that overcomes the limitations of the current objective functions designed for RPL. They also present an extension version of RPL that supports mobility to provide a better quality of service in dynamic networks application [30] presents GeoRank, a geographic routing approach for the RPL. It integrates RPL protocol with the position-based routing protocol make it possible to find short routes in different conditions. It avoids the use of DAO

(Destination Advertisement Object) control messages required in RPL which makes it more scalable than standard RPL. In [31] first the unreliability issues of standard RPL is investigated, then a new implementation of the RPL standard for the Contiki operating system which improves data delivery reliability is presented. The designed protocol provides simple routing optimizations, enhanced link estimation capabilities, and efficient management of neighbor tables.

Dynamic network control approach based on software defined networking for meeting the specific communication requirements of both distribution and transmission power grid has been analyzed in [32]. In [33] SDN has been utilized for automatic fail-over mechanisms, load balancing, and quality of service guarantees in smart grid networks. An architecture based on OpenFlow protocol that establishes different types of flows in IEC61850 has been developed in [34]. Three potential use cases including, (1) Enhanced data exchange, (2) Virtual network for distributed energy resources aggregation and (3) Smart building energy management are investigated in [15]. It has been proven that SDN is successful to develop a centralized routing and traffic engineering protocol for any kind of communication networks.

In [35] the SDN and Mobile Social Networks (MSN) is integrated together to improve the traditional MSNs and eliminate some problems. The proposed method which is called SDMSN presents a new framework design and flow table based on the social features including social ties, community, and mobility. The data flow can be operated with social consideration. Also, a new social switch based on social degree and social stay time is proposed, which better deliver the content among mobile users and reduce both control traffic and data traffic.

3 The proposed model

3.1 SDN-enabled architecture

The proposed system architecture is depicted in Fig. 2, which consists of two different parts: the SDN controller and SDN data plane. The SDN controller utilizes the multi-class RPL service and network operating system. The multi-class RPL service, which uses the REST API to connect to a network operating system such as NOX, is an intelligent application based on the OpenFlow protocol. It creates a logically centralized control plane to build a global view of the physical topology of all SDN-based nodes in the network connected to the same SDN controller. This centric design helps to develop an optimized routing protocol to support different classes of traffic. As we have a consistent view of the network state and application-aware routing, it will be possible to build the optimum DODAG. This implementation reduces the complexity of control interactions in order to distribute the network states and other related information in the network. The SDN-enabled architecture consists of the following components:

- **Network Link Discovery:** This unit is responsible for discovering and maintaining the status of all physical links in the network using traditional link layer discovery protocols such as IEEE 802.1AB. Using link layer discovery protocol, each node advertises its identity, its resource and traffic status and a list of its neighbor nodes. After that, all network nodes send their gathered information to the controller to be stored in the proper database for further processing.
- **Topology Manager:** This unit builds and maintains the global network topology. It utilizes the link information already discovered and stored in the database by the network link discovery unit.
- **Virtual Routing:** This unit is responsible for finding the virtual DODAG and the best parents and the route for all network nodes. It provides the interoperability between the SDN controller and the other SDN nodes in the network. It determines the route's adjacent neighbors and finds the next hop for each network node.

At the SDN data plane, there are some SDN-based network nodes which support OpenFlow protocol and have an OpenFlow client module. Each node utilizes its flow table to forward the incoming packet to the proper next hop. Note that as the proposed architecture can support multi-class traffic, there are different entries for each traffic class of each flow.

As shown in Fig. 3, each SDN node needs to send the control purpose traffics such as the route setup requests for new flows and traffic status, to the SDN controller. Based on the continuously received control messages, the controller optimizes the routes for data flow according to dynamically changing traffic patterns and flow QoS requirements, and sets up the forwarding tables of each SDN node along the optimal path via certain secure protocols (e.g., OpenFlow), thus enabling highly efficient data transmissions and superior link utilization.

3.2 OMC-RPL design

In this section, we explain the proposed OMC-RPL protocol, which supports data classification. As we already mentioned in the related work section, the existing protocols that offer the QoS by using RPL, usually face with two significant shortcomings:

1) Most of the approaches do not provide a comprehensive and holistic objective function. For example, an objective function may improve the end-to-end delay by finding the most proper path toward the sink, but as all packets try to use the same path, it causes negative effects on energy consumption.
2) The data classification, which is one of the most important requirements for assuring the QoS, is not supported by the available methods. The main reason is that if we categorize the data, each class type has its own specification, and it should be treated in a distinctive way. It means that we need an objective function supporting different classes of

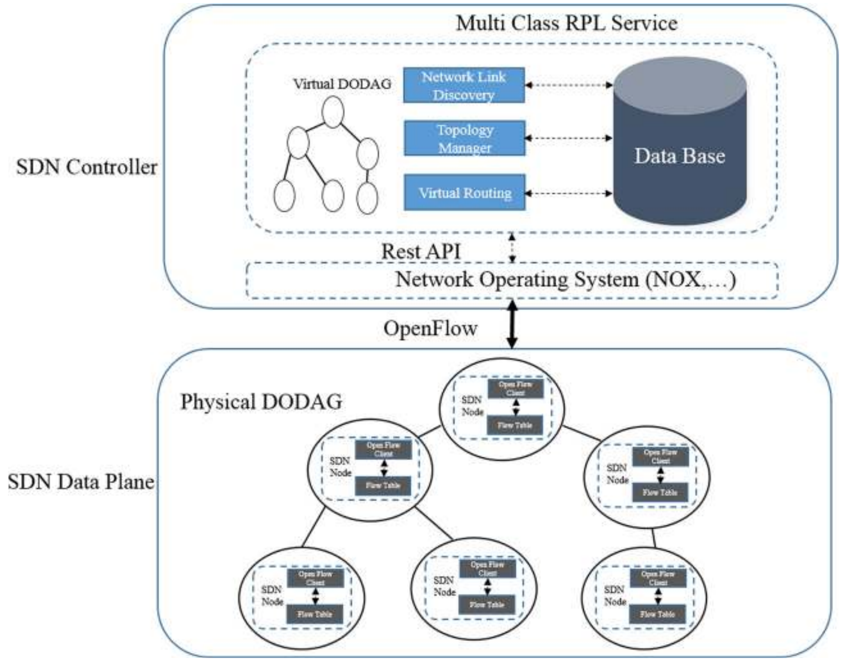


Fig. 2 The system architecture

data, which is a significant challenge. Some studies use two different objective functions for multiple instances of RPL. In fact, some networks may run multiple instances of RPL concurrently, but logically these instances are different and independent. We need an approach to make the data classification applicable to a single instance of RPL with various requirements.

The proposed OMC-RPL protocol operates as follows: at the first step, the network link discovery module discovers the network nodes and their related information. It then discovers the virtual topology of the physical network. At the next step, the virtual DODAG should be discovered. To better understand the virtual DODAG construction process, we provide a flowchart in Fig. 4, which shows the steps of creating the virtual DODAG regarding just two classes of data. Note that all messages are converted to the OpenFlow protocol format and forwarded to the specified node. To construct the virtual DODAG, the root node, as the first parent, starts the operation and investigates the neighbor's nodes, which are in its range. If no rank is assigned to the neighbors, the algorithm calculates two ranks for the node for each traffic class using objective functions and creates a list which includes pairs of (rank1, parent) and (rank2, parent). The nodes with newly calculated ranks repeat investigating their neighbors. Consequently, if a node already has a rank, the algorithm performs steps to decide if it could find a better parent with lower rank. Note that in the DODAG; each node can have several parents, one as the

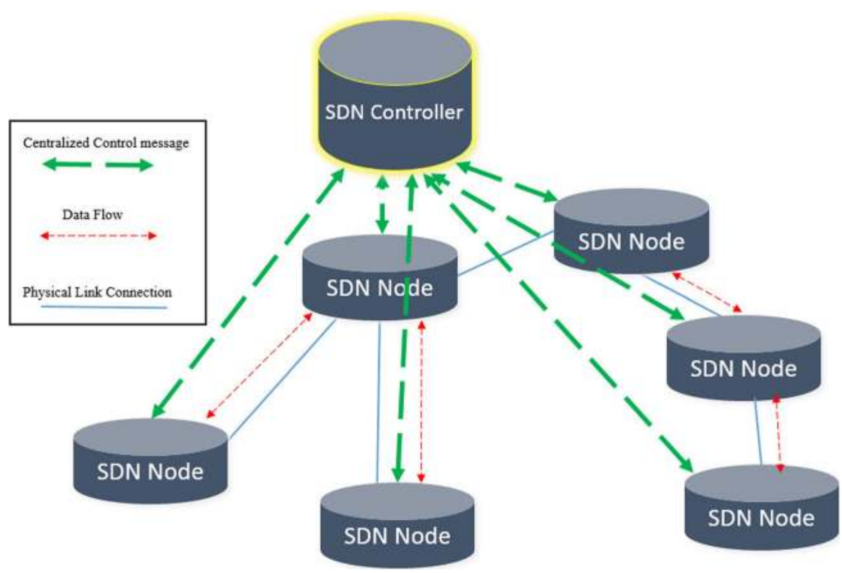


Fig. 3 Centralized control traffic and data flow in the system

preferred parent and the others as the replacements in the case of failure. Furthermore, in the proposed method, each node may have different preferred parents proportional to the number of classes. Each parent is only suitable for its relevant class. Steps in which the proposed algorithm chooses the preferred parents are as follows:

The neighbor node checks the parent rank for each class, and if it is not greater than the current rank, then it is reasonable to find the new parents otherwise ignore it. The neighbor node computes the new ranks based on the fresh information to see if it is really from a more suited parent. If so, it updates the pairs of (rank and parent) in its list and removes the parents with the greater rank. To clarify the concern of overhead, we should note that there are not separate routing tables, we just keep different ranks and preferred parents for each node.

When the virtual DODAG is constructed, the upward route is clear because the only routing process for the intermediate nodes is to send their packets, based on their class, to the preferred parents, which are chosen during the DODAG's construction procedure. We introduce a data traffic classifier which associates network traffic flows with the application that generated it. The classification is performed at each network node based on the application. The SDN controller utilizes a global traffic learner to be aware of the traffic classes used in the network. The traffic classifier is responsible for identifying the QoS class of a traffic flow through a mapping function. This is simply a function that takes a few features of the traffic flow as the inputs and gives the QoS class of the traffic flow as the output. Figure 5 shows the location of the traffic classifier and the global traffic learner.

Algorithm1 is the generalized OMC-RPL protocol for *n* classes of data, which is run by the SDN controller.

Algorithm 1: OMC-RPL Algorithm Based on *n* Classes Runs in SDN Controller

```
Input:
n classes of data
Objective Function (which is function of node congestion, link congestion and class of
data)
Output:
List of {rank_1- parent, rank_2-parent … rank_n-parent} for each j ∈ {1… x}, where x is the
number of nodes
Method:
(1) Root as a first parent start the algorithm and investigate the neighbors
(2) Repeat
(3) If (no rank is assigned to the neighbors)
        {
        for (i = 1 to n)
                {
                Compute rank_i for node j based on OF
                Create list (rank_i, parent)
                }
        Using new ranks, investigate the neighbors of current node as a parent
        }
(4) else
        for (i = 1 to n)
        {
        if (parent rank_i < current rank_i)
                Compute new rank_i
                if (new rank_i < current rank_i)
                        {
                        Change the preferred parent for class i
                        Drop the parents whose rank_i is greater than new rank_i
                        }
                else
                        Assume the parent node as reserved parent in a case of failure
        }
(5) until all nodes attach to DODAG
```

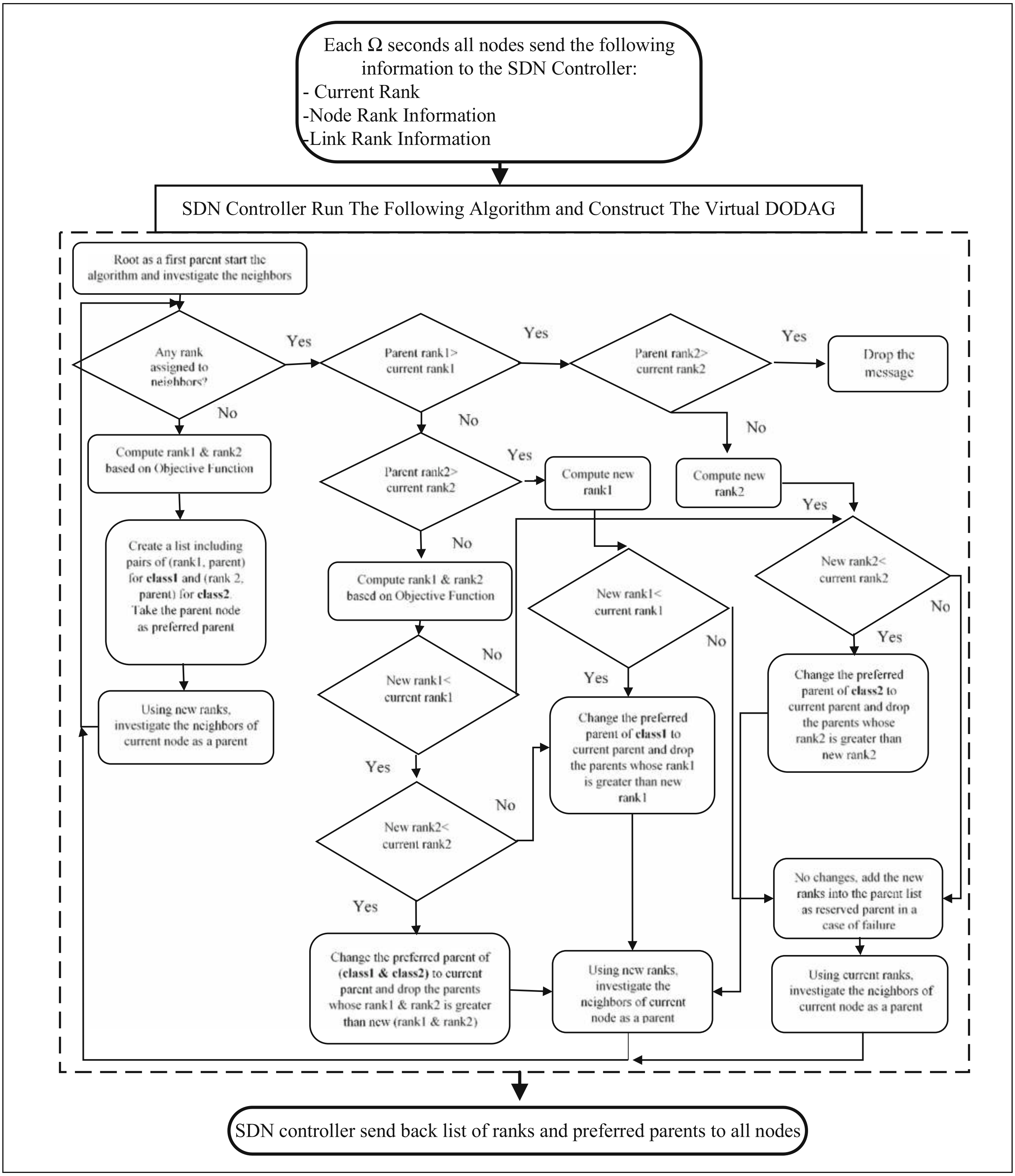


Fig. 4 The OMC-RPL algorithm for constructing virtual DODAG with two classes

We have investigated the worst and best case scenarios, to compute the complexity of the Algorithm 1. Regarding the main repeat-until loop in the algorithm and the number of exchanged messages between the nodes, the worst case happens when all nodes connect to each other, and make a completed graph. The best case occurs when the deployment of nodes is shaped as a line, and each node connects to two nodes at most. Based on the node deployment, Fig. 6a and b show the worst and the best case scenarios, respectively. At the worst case, the repeat-until loop repeats $x^2 - x$ times and at the best case it

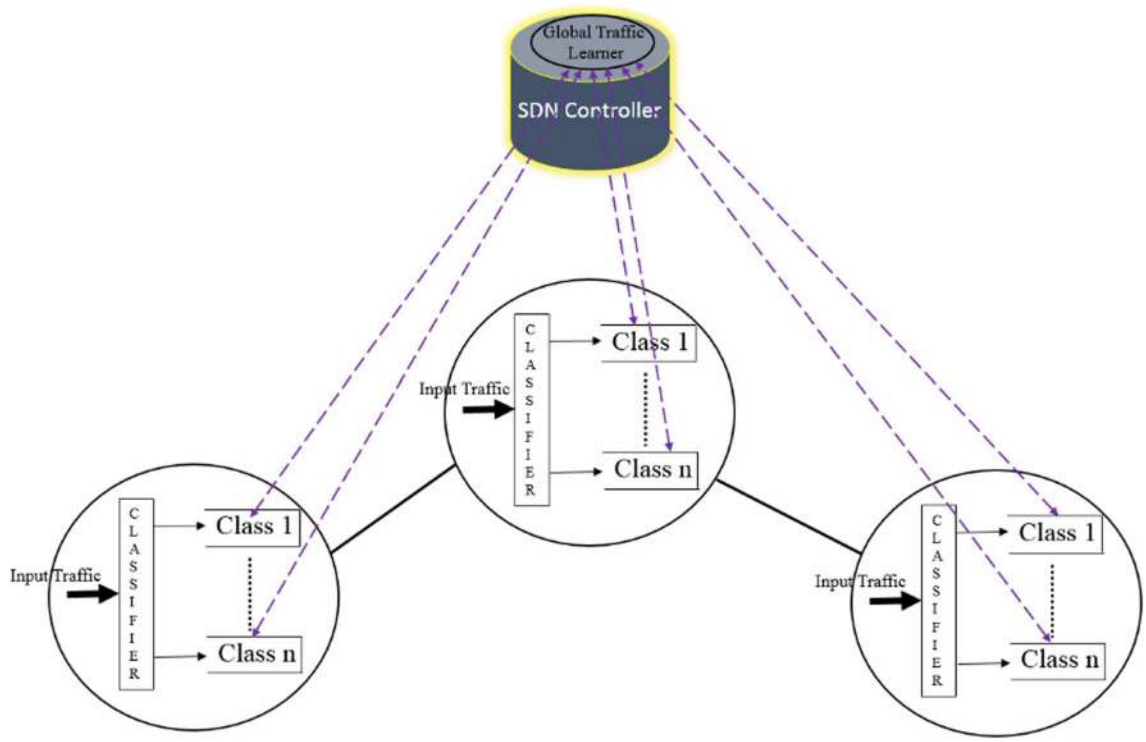


Fig. 5 The proposed traffic classifier

repeats $2(x-1)$ times, where x is the number of nodes. Inside the repeat-until loop, there is an if-else structure that in each repeat either the "if" or the "else" is run n times, where n is the number of classes. Therefore, the complexity of Algorithm1 at worst and best cases are given in Eqs. (1) and (2) as follows:

$$\textbf{\textit{Worst case}} : T(x) = \left(x^2 - x\right) \times n = nx^2 - nx = O\left(nx^2\right) \quad (1)$$

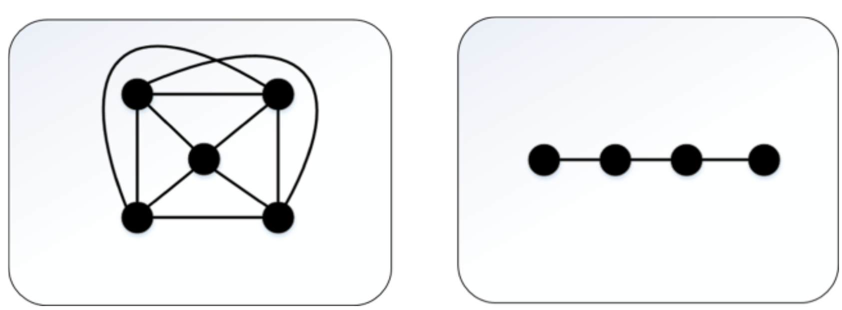
$$\textbf{\textit{Best case}} : T(x) = 2(x-1) \times n = 2nx - 2n = O(nx) \quad (2)$$

As the number of classes in our problem is a fixed and small value, then we can omit n from the complexity. So, the worst and the best complexity of Algorithm1 is equal to $O(x^2)$ and $O(x)$, respectively.

We provide an example to describe the proposed algorithm better. Figure 7a shows a sample network of twenty nodes and two roots in the AMI network. Each node may be used to carry traffics of different applications such as Demand Response (DR), AMI, smart street light and smart home devices with various QoS requirements. The root could be a concentrator in a station. After performing the OMC-RPL algorithm in the central controller, a virtual DODAG, shown in Fig. 7b is created. It is evident that this exemplary DODAG is formed based on the specific objective function. If we change the objective function, we will probably face another DODAG. As mentioned earlier, according to two classes of data, each node has two preferred parents (which could be different from each other), and also, each node could have none or several reserved parents.

3.3 The proposed class based objective function

Now, the challenging issue is designing an objective function to support different classes of data. The objective function should provide the low end-to-end delay for real-time traffics,



(a) Worst Case

(b) Best Case

Fig. 6 Computing the complexity of the Algorithm1 based on node deployment

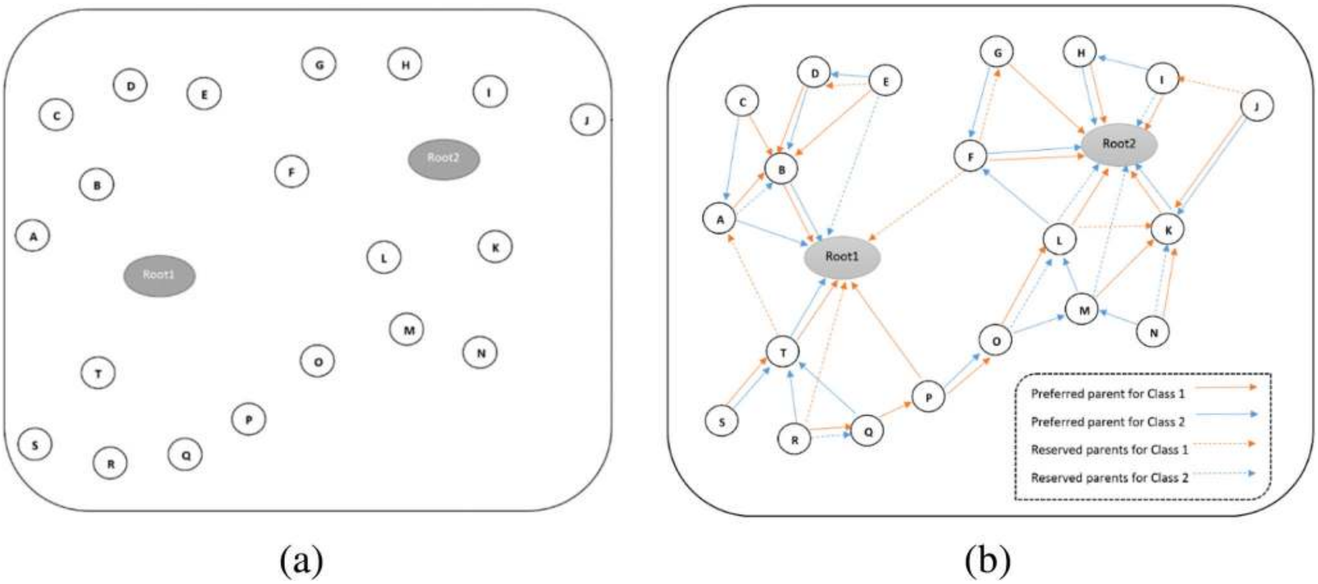


Fig. 7 **a** A sample network of nodes of various applications in HAN and NAN **b** An exemplary DODAG after using OMC-RPL based on two classes of data

high packet delivery ratio and low packet loss probability for the loss sensitive applications. Therefore, our goal is to propose a comprehensive objective function that satisfies these properties. Instead of having n objective functions, we suggest weighting parameters that make the difference for each class of data based on its requirements. The main components of the proposed objective function are the Node Congestion (NC), the Link Congestion (LC) and the Propagation Delay (D). The remaining energy is not an important metric in most of the smart grid applications. However, some studies focus on the role of wireless sensor networks in the smart grid [36–40]. Therefore, in some particular cases, the remaining energy may be an important metric, but in general it is not. To address the above both cases, the proposed objective function is designed in a way that the Remaining Energy (RE), can be intentionally considered or removed from the objective function. The proposed objective function is given as follows:

$$R_n^i = R_n^{p_i} + \frac{\alpha_n\left(NC^i + D^i\right) + \beta_n LC^i}{1-\theta\left(1-RE^i\right)} + 1 \quad (3)$$

The terms of the above objective function are defined in Table 1. The α_n and β_n are weighting parameters, which are used to control the effectiveness of NC, LC and D, respectively. These parameters are chosen based on the class type. The parameter of θ is used to make a decision about the existence of the RE parameter. If energy is an important metric, then θ is set to 1, otherwise, it is set to zero. When the remaining energy is not an important metric, the proposed objective function is given as follows:

$$R_n^i = R_n^{p_i} + \alpha_n\left(NC^i + D^i\right) + \beta_n LC^i + 1 \quad (4)$$

Note that for real time services which delay is an important metric, node congestion and distance are more important than link congestion, while for assured services which low packet loss is important, the link congestion is an important metric. Distinctive types of traffic and five classes of data related to LLN have been presented in [41]. Each data class faces with different QoS requirements. In the proposed objective function, by changing the weighting coefficients, it is possible to satisfy the required QoS. In fact, the data classification is based on the value of the allowed delay times for various types of applications. For example, for the critical and real-time

Table 1 Definition of parameters used in the objective function

Parameters	Definition
R_n^i	Rank of node i for n ' th class
$R_n^{p_i}$	The rank of preferred parent for n ' th class in node i
α_n	Weighting coefficient for n ' th class between (0–1)
β_n	Weighting coefficient for n ' th class between (0–1)
NC^i	The node congestion for node i
LC^i	The link congestion of node i and its neighbor
$\boldsymbol{D^i}$	Propagation delay of node i and its neighbor
RE^i	The percentage of remaining energy in node i
θ	Coefficient equal to 0 or 1
γ	Fixed coefficient between (0–1)
λ^i	Arrival rate at node i
μ^i	Service rate at node i
Q^i	Instant queue length of at node i
L^i	Buffer length at node i
m^{ij}	data packets transmitted from node i to node j in Ω seconds
s^{ij}	Number of successful network layer transmission between node i to node j in Ω seconds
d^{ij}	The distance between node i and node j
V	The propagation speed

traffics, the end-to-end delay should be very low. These types of traffic are categorized to the high-priority traffic class. Assume a range of applications from real-time to non-real-time, which can be divided into several classes; the first class includes the most critical application, and the last one includes the least important. When there are real-time packets, the values of weighting parameters should be selected in a way that the objective function leads to the best possible path (appropriate nodes and links that minimize the end-to-end delay and improve the data delivery rate) towards the root.

The node and link congestions, and propagation delay are determined as follows:

$$NC^i{}_{current} = \rho^i \times \omega^i \tag{5}$$

$$NC^i = (1-\gamma)NC^i{}_{old} + \gamma NC^i{}_{current} \tag{6}$$

$$\rho^i = {}^{\lambda^i}\Big/_{\mu^i} \tag{7}$$

$$\omega^i = {}^{Q^i}\Big/_{L^i} \tag{8}$$

$$LC^i = 1-\left({}^{s^{ij}}\Big/_{m^{ij}}\right) \tag{9}$$

$$D^i = d^{ij}\Big/V \tag{10}$$

Equation (5) computes the node congestion by multiplying traffic load (ρ^i) by buffer occupancy (ω^i). To prevent the instant fluctuations and to keep the history of congestion in the node, Eq. (6) is used. When the traffic density or the buffer occupancy is high, the NC is also high. Equation (9) computes the link congestion by counting the proportion of packets that are delivered successfully to the destination to the number of all sent packets in **Ω** seconds. Equation (10) computes the propagation delay by dividing node distance by the propagation speed (V). As at each node i for any particular traffic class n, the node rank (R_n^i) is directly related to the node/link congestion and propagation delay levels (NC^i, LC^i and D^i), when there is a congestion at the node or at the link, the NC^i, LC^i and D^i are increased which rises the node rank. As mentioned earlier, the algorithm chooses the node with the lowest rank as the preferred parent, then, by decreasing any congestion at the node or the link, the node rank is decreased, which increases the chance of choosing that particular node as the parent node.

As discussed earlier, in comparison with the non-real-time classes, the critical and sensitive classes of data need the path with the lowest end to end delay; with regard to this issue, it is reasonable to find the path that includes the nodes with the lowest node congestion and propagation delay. As a rule, it means that for the critical classes, we should keep the α_n coefficient greater than β_n, and for the less important classes, it happens vice versa. This classification idea causes the spread of traffic through the network, which leads to congestion prevention.

In order to have distinctive weighting coefficients for each class of data, and concerning the rule mentioned above, we need specific ranges for the coefficients. The ranges for the weighting parameters, equivalent to the number of classes, is depicted in Fig. 8. For example, in a case of two classes of data, the optimized NC and D coefficient (α_1) for class1 (critical class) should be found in the interval of 0.5 to 1, and the optimized LC coefficient (β_1) in the interval of 0 to 0.5. For class 2 (unimportant class), α_2 and β_2 should be chosen in the range of 0 to 0.5 and 0.5 to 1, respectively.

Regarding the specific ranges, the best values for the weighting coefficients are found by using the PSO (Particle Swarm Optimization) algorithm. PSO is a population-based optimization algorithm that is formed based on the social behavior of a swarm of birds and fish looking for food [42]. Searching rules in this algorithm are easy and yet meaningful. The computation time is low, and there is no need for much memory space; these advantages make it popular in our target field. The procedure of the PSO algorithm in finding optimal values follows the behavior of animal society by using the best personal/global experience of particles. PSO consists of a swarm of particles, where each particle represents a potential solution [43]. Although recently, there have been several modifications from the original PSO, the main idea of PSO says that a velocity vector influences the position of the particle toward the optimized answer. Let $x_i(t)$ denote the position of particle i in the search space at time step t (denotes discrete

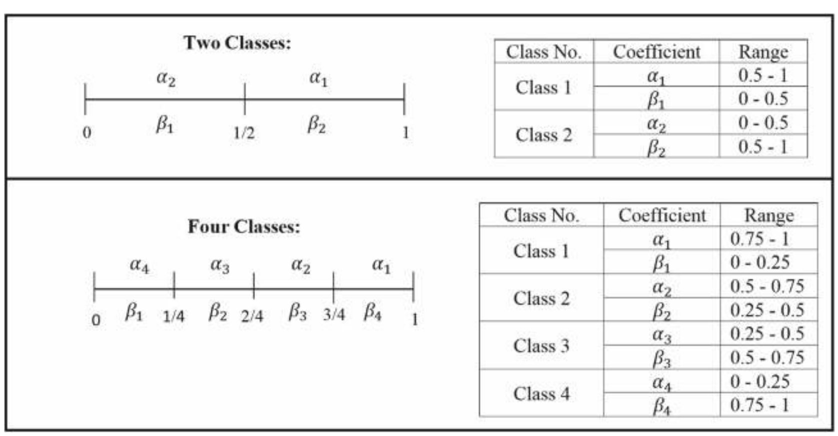


Fig. 8 Specified range of weighting coefficients for two and four traffic classes

Table 2 The traffic classes characteristics

Traffic class name	Traffic characteristics	Required delay	Required packet loss	Sample application
Class 1	Packet Size: Small Rate: Periodic, Few Packets	Very low	Medium-High	Teleprotection
Class 2	Packet Size: Small Rate: Periodic, Few Packets	Low	Medium	Synchrophasors
Class 3	Packet Size: Big Rate: Variable	Low-Medium	Low	AMI
Class 4	Packet Size: Big Rate: Variable	Medium-High	Very Low	SCADA

time steps). According to Eq. (11), the new position of the particle is obtained by adding a velocity $v_i(t)$ to the current position. The velocity is calculated based on the Eq. (12), which is the outcome vector of the previous velocity and local best and global best values [44].

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{11}$$

$$v_i(t) = v_i(t-1) + \ c_1 r_1 (localbest(t) - x_i(t-1)) + c_2 r_2 (globalbest(t) - x_i(t-1)) \tag{12}$$

In Eq. (12), *localbest*(t) and *globalbest*(t) respectively show the best personal and the best neighborhood experience of particles in time slot t. $c_1 and\ c_2$ are acceleration coefficients, and $r_1 and\ r_2$ are random numbers between 0 and 1. After a certain number of repetitions, the algorithm will find the optimized answers in the search space.

Eventually, the PSO procedure of finding the optimized values for weighting parameters is as follows: the required initial values are determined; the random weighting coefficient values are selected for the particles; the PSO objective function, which we consider the average end-to-end delay, is implemented; then the best personal and global experiences are identified. The algorithm is repeated based on the Eq. (11), until finding the optimized values.

The weighting parameters are determined in an offline mode before the main simulation starts. Therefrom, the fixed amounts of the coefficient are used in the objective function. Note that; this doesn't mean that we miss the dynamism of the network, because the parameters of the Node Congestion, Link Congestion, Propagation Delay and Energy are continuously updated, and the constructed DODAG is based on this information.

Table 3 General information about the simulation environment

Simulator	Riverbed 18.0
Simulation area	250×250 m^2
Number of nodes	99
Number of concentrators	1
Test duration	1 h

4 Performance evaluation

In this section, using the computer simulation, the performance of proposed OMC-RPL is investigated and compared with the traditional RPL algorithm. For this purpose, a sample network is created, and several nodes are deployed randomly. We implement the OMC-RPL supporting four different classes. Therefore, different traffic types belonging to each class of data should be generated by the various applications. To achieve this goal, four different applications in accordance with four classes of data are randomly assigned to each node. Then, the nodes generate data packets and tag them with the relevant class. The traffic characteristics of each class are given in Table 2. Table 3 shows the general information about the simulation environment.

Before evaluating the performance of the proposed method, we need to find the optimized values for the weighting coefficients. The above-described PSO has been implemented in the simulation software. The achieved values of the parameters for two and four classes of data regarding the Eq. (4), is shown in Table 4. For example, in the case of two classes of data and by using the achieved values, the objective functions without considering the energy parameter for the first and second class are given as follows:

$$R_1^i = R_1^{p_i} + 0.81\left(NC^i + D^i\right) + 0.34 \times LC^i + 1 \tag{13}$$

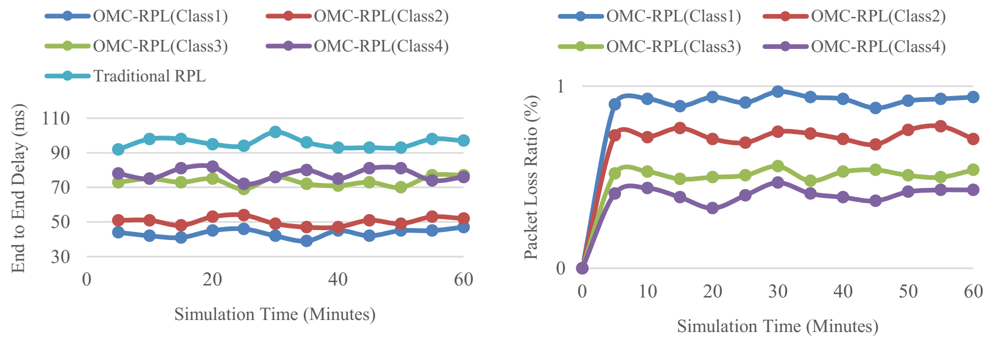
$$R_2^i = R_2^{p_i} + 0.43\left(NC^i + D^i\right) + 0.78 \times LC^i + 1 \tag{14}$$

In the following subsections, the evaluation results are presented for both OMC-RPL and SDN controller. In the first part, by using the objective function of Eqs. (3) and (4), two

Table 4 The values of weighting coefficients

Two classes	$\alpha_1 = 0.81$	$\beta_1 = 0.34$
	$\alpha_2 = 0.43$	$\beta_2 = 0.78$
Four classes	$\alpha_1 = 0.78$	$\beta_1 = 0.20$
	$\alpha_2 = 0.69$	$\beta_2 = 0.42$
	$\alpha_3 = 0.31$	$\beta_3 = 0.77$
	$\alpha_4 = 0.19$	$\beta_4 = 0.88$

Fig. 9 Comparison of end-to-end delay and packet loss (all nodes work normally)



(a) The end-to-end delay (b) Packet loss among the Classes of OMC-RPL

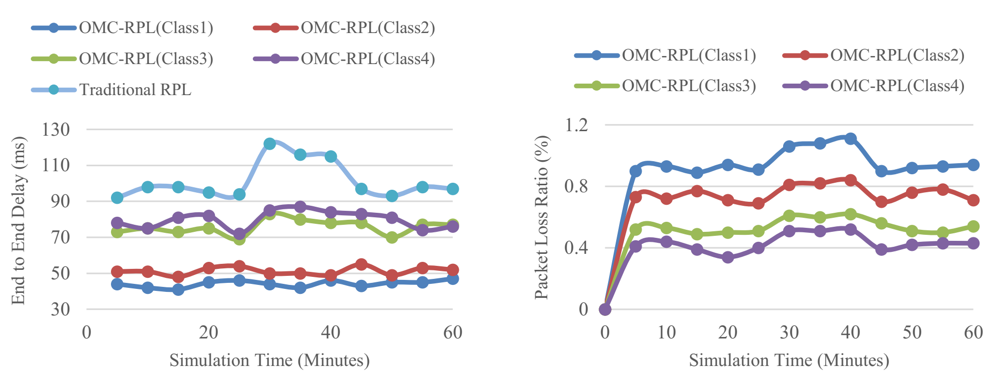
important factors of the smart grid communication network including the end-to-end delay and packet loss probability are investigated. However, as one of our contribution in this paper is to present a holistic objective function for the smart grid applications, for those applications which leverage the wireless sensor network technology, we consider the effect of remaining energy in the objective function given in Eq. (3) and evaluate the energy consumption performance. For the SDN part of proposed method, by evaluating the number of transmitted packets and the total energy consumption, we will show how the model can utilize the benefits of a centralized controller of software-defined networking to achieve better performance.

4.1 Performance of OMC-RPL

In this subsection, we evaluate the performance of OMC-RPL in distinctive scenarios. In all scenarios, the OMC-RPL with four classes of service and the traditional RPL, which is based on ETX, are used for comparison. In the first section, by setting $\theta = 0$, we evaluate the end-to-end delay and the packet loss performance of both traditional RPL and the proposed OMC-RPL. The results shown in Fig. 9a, confirm that the classification idea outperforms the RPL with single objective function. Using OMC-RPL, as expected, Class1 has the least end-to-end delay and orderly Class2, 3, 4 experience higher delays, which is compatible with our classification idea. As seen in Fig. 9a, the results for Class1 and Class2 are close to each other. The main reason is that the characteristics of traffics for these two classes are, to some extent, similar to each other; this issue also applies for Class3 and Class4. During the simulation time, in some points, Class3 and Class4 are experiencing the same delay. Although the performance of OMC-RPL for Class1 and Class2 is completely acceptable, the supremacy of Class3 is negligible rather than Class4 and even in some points class3 is worse than class4.

In the same scenario, Fig. 9b depicts the comparison of packet loss among the classes of OMC-RPL. As we expect, according to required information in Table 2, Class4 has the least packet loss, and the other classes experience higher amounts of packet loss respectively. Results confirm that during the simulation time, the packet loss of each class is less than one percent. The average packet loss ratio of Class1, 2, 3 and 4 are 0.923%, 0.729%, 0.517% and 0.405%, respectively. The diagram shows that the level of packet loss in different classes are so close to each other. Although Class1 has the least end-to-end delay, it experiences the most packet loss, which shows that the weighting coefficients work properly. In fact, our proposed objective function provides a path with the lowest delay for the traffics of the first class, and also the lowest packet loss for the traffics of the last class.

Fig. 10 End-to-end delay and packet loss (some nodes are congested in time interval [30, 40])



(a) The end-to-end delay (b) Packet loss among the Classes of OMC-RPL

In the second scenario, we evaluate the end-to-end delay in case of sudden congestion. In this case, some nodes are congested randomly during the time interval [30, 40]. The results are depicted in Fig. 10a. The peaks in the traditional RPL clearly show the occurrence of congestion. Results confirm that in all cases, the OMC-RPL acts more efficiently than the traditional RPL. In the OMC-RPL, the process of DODAG construction is continuously repeated. Any changes in nodes or links lead to modifications in ranks and create a new DODAG. In the RPL, any changes in nodes are no matter and cause increased drop rates in case of congestion. That is why, at these moments, the end-to-end delay increases slightly in the OMC-RPL and rises sharply in the RPL. The increase of delay in Class1 and Class2 is negligible since in Class3 and Class4 we can observe a smooth increase. It means that, in cases of congestion, the OMC-RPL is more effective for critical classes.

Also, in this scenario, the packet loss among four classes of OMC-RPL is evaluated and presented in Fig. 10b. As it past half of the simulation time, the packet loss increase because of the congested nodes. Class4 outperform the other classes in respect of packet loss. Regarding the chart, it is evident that the increase in Class4 and 3 is smoother than the Class2 and 1. Because of the congested nodes, the maximum packet loss occurs for Class1 at time 40, which is 1.11%. Except the congested time interval of [30, 40] the packet loss ratio is less than one percent.

In the next scenario, we suppose that some network nodes are failed. Figure 11a depicts the end-to-end delay when some nodes have been failed. The failure occurs in the thirtieth minutes of the simulation. When the failure happens in RPL (ETX), the end-to-end delay increases significantly, while this growth in Class1 and 2 of OMC-RPL is negligible, and in Class3 and 4 is smooth. We know that when a node fails, the algorithm quickly uses the reserved parents. The proposed OMC-RPL, even at the time of node failure manages the flow of packets by using different objective functions and keeps a reasonable delay for the critical classes. Figure 11b shows the packet loss in this scenario. As the node failure happens, the average packet loss ratio of Class1, 2, 3 and 4 increase to 1.014, 0.776, 0.579 and 0.472%, respectively. Even by missing some nodes, Class4 and 3 keep a reasonable amount of packet loss, which is our goal for the applications that belongs to these classes.

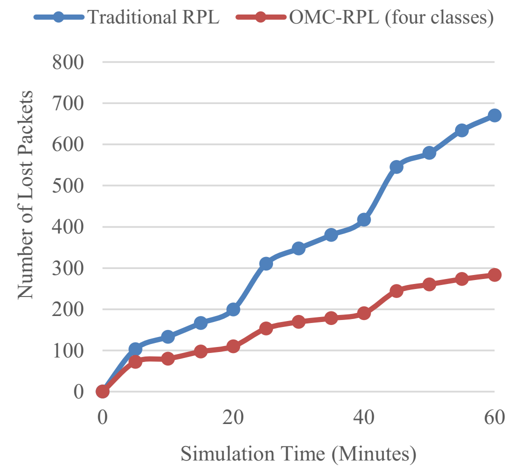
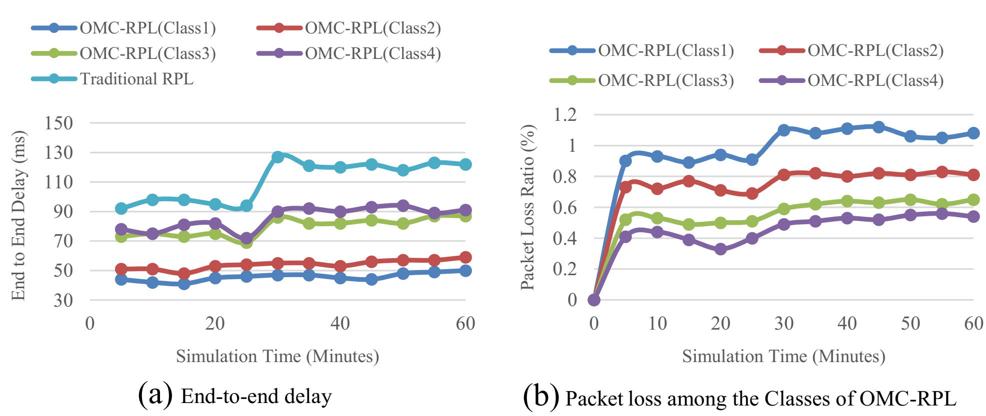


Fig. 12 Comparison of the packet loss

The results of end-to-end delay evaluation under different circumstances show that the OMC-RPL notably decreases the delay for all classes of service, and especially the critical one, which is an important issue. Also, by using the proposed objective function, the packet loss rate based on the traffic requirements of each class is much less than the Traditional RPL.

The following scenario evaluates the packet loss performance of OMC-RPL and traditional RPL in general. Figure 12 shows the total number of lost packet versus simulation time. It confirms that the objective function of the traditional RPL is not proper, and this method experiences more packet loss than OMC-RPL. At the end of the simulation, the packet loss of the traditional RPL is two times more than that of OMC-RPL. The comprehensiveness of the proposed objective function avoids some of the packet losses.

Fig. 11 Comparison of the end-to-end delay and packet loss (some nodes fail at time 30)



In the last scenario, we evaluate the average node's queue size by choosing four sample nodes. Figure 13 shows the average queue size in the selected nodes during the simulation time. It can be seen that when RPL is utilized, a node like node 2 is too busy, while some other nodes, such as node 1, are rarely chosen as the preferred parent and are always idle. When using OMC-RPL, the average queue size does not exceed more than half of the capacity. In fact, OMC-RPL, by using a holistic objective function and various parameters, balances the traffic load in the network.

The results confirm that the classification idea in OMC-RPL and the proposed objective function can support the appropriate routes for any traffic class.

In the following scenario, for those smart grid applications which leverage the wireless sensor network technology, by setting $\theta = 1$, we consider the effect of remaining energy in objective function given in Eq. (3) and evaluate the network lifetime. In this case, half of the nodes work with battery. We define the network lifetime as the time until the first node (among the nodes that work with battery) in the network runs out of energy. Simulation results confirm that the lifetime of the proposed OMC-RPL with four classes and the traditional RPL are equal to 5515 s and 3293 s, respectively. This supremacy shows the effectiveness of data classification and the proposed objective function. In another scenario, we investigate the fairness in the energy consumption. If the energy consumption among all nodes is fair, then the best network performance could be achieved. In order to compute the percentage of the fairness we use the following equations:

$$\mathrm{DEV} = \sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{energy}-\mathrm{Ave})^2 \tag{15}$$

$$\mathrm{Fairness} = 1-\frac{\mathrm{DEV}}{\mathrm{DEV}_{\mathrm{worst}}} \tag{16}$$

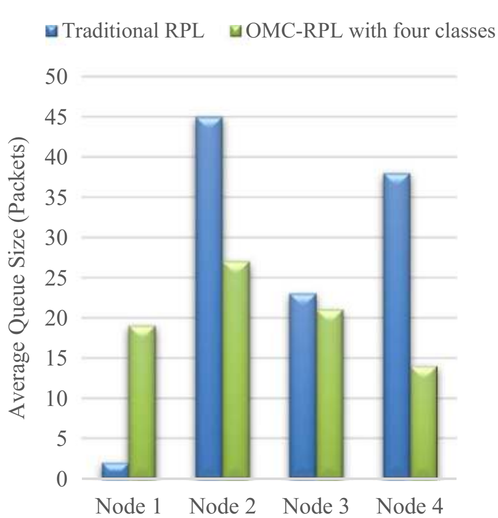


Fig. 13 The average queue length at different nodes

In the above equations, "energy" shows the remaining energy in each node, and "Ave" and "DEV" are the average and standard deviation of the remaining energy for all nodes. "$\mathrm{DEV}_{\mathrm{worst}}$" is the worst standard deviation, which is achieved when half of the nodes have the highest amount of energy and the other half have zero energy. According to the above description, the fairness in energy consumption of OMC-RPL and RPL are computed, which is equal to 59 and 41%, respectively. It is clear that the fairness of OMC-RPL is 18% more than RPL. This fairness leads to prolonging the network lifetime, which is proven in the previous scenario.

4.2 Incorporation of OMC-RPL with SDN

In this section, to investigate the role of the SDN central controller on the performance of the proposed model, the following scenarios are presented. In the first scenario, the required number of packets to create a DODAG is evaluated. The results are given in Table 5. The first two rows of the table indicate the number of required packets to construct the DODAG in the proposed OMC-RPL and traditional RPL without using the idea of the SDN central controller. Each time that a DODAG is created or reconstructed, for both the traditional RPL and OMC-RPL (without the central controller), 503 packets are exchanged among all nodes, while, by using the SDN central controller, the required number of packets is just 196. Note that when using the SDN concept, we assume that each node can send the required information to the SDN central controller just by one hop.

In the next experiment, OMC-RPL (with and without the central controller) and the traditional RPL are evaluated in terms of required bandwidth and energy consumption. Equations (17) and (18) present the energy consumption model for packet exchange over the wireless link. Equation (17) gives the amount of energy consumed by node m to transmit a packet of length x bits over the link (m, n). Equation (18) computes the energy consumed by node n to receive the packets from node m. The definition of the parameters used in these equations is given in Table 6.

$$E_{m,n}(x,\ r) = \left(P_t + \frac{P_{m,\ n}}{k}\right)\frac{x}{r} \tag{17}$$

Table 5 The number of required packets to create a DODAG in different cases

Methods	Number of required packets
OMC-RPL (without central controller)	503
Traditional RPL	503
OMC002DRPL (with central controller)	196

Table 6 Definitions of the parameters used in energy consumption model

Parameters	Definitions
$(m\ ,\ n)$	Wireless link between sender m and receiver n
r	The rate of data transmission over the physical link of $(m\ ,\ n)$
P_t	Required power to run the processing circuits of the transmitter
P_r	Required power to run the processing circuits of the receiver
$P_{m\ ,\ n}$	The transmission power from m to n
k	Efficiency of the power amplifier

$$W_{m,n}(x,\ r) = \frac{P_r}{r}x \tag{18}$$

Figure 14 shows the comparison of the bandwidth and energy consumption of OMC-RPL (with and without a central controller) and the traditional RPL. In both Fig. 14a and b, the sharp growths in the diagram express the DODAG construction process, which needs many packets to be exchanged. Accordingly, it causes the increase in bandwidth and energy consumption. In Fig. 14a, when the DODAG construction occurs, the OMC-RPL with central controller experiences the least bandwidth consumption. Except the times of DODAG construction, all compared methods, have a almost the same bandwidth consumption. Generally, because of the smaller packet size in the traditional RPL, its bandwidth consumption in less than the other approaches. As shown in Fig. 14b, the required energy to construct a DODAG with OMC-RPL (without the central controller) and the traditional RPL are 4377 and 4124 mJ, respectively. It means that the traditional RPL consumes a little bit less energy than OMC-RPL (without the central controller). This is because OMC-RPL's messages need some fields to carry the traffic class identifier. It is clear that OMC-RPL (with the central controller), by omitting the unnecessary packet exchanges, significantly decreases the energy consumption and compensates this problem.

5 Conclusion

The Routing Protocol for Low-Power and Lossy Network (RPL) has become one of the main routing protocols in the NAN networks of the smart grid communication infrastructure. Lots of research has been done on this issue. In this paper, by utilizing the benefits of virtualization and software-defined networking, we presented OMC-RPL (Optimized Multi-Class-RPL) which is a modified class-based version of RPL and can support different traffic classes. OMC-RPL is run centrally in the controller. Since data classification is the main requirement of providing QoS, the proposed model, with comprehensive objective function regarding the QoS metrics, can support multi-classes of data. In this regard, the PSO optimization algorithm was used to find the best values of coefficients that are used in the objective function. The OMC-RPL with four classes and various scenarios was simulated; the results in comparison with traditional RPL are significant. OMC-RPL decreases the end-to-end delay, balances the traffic load in the network and reduces the packet loss rate. Then OMC-RPL, by using the concept of SDN, was implemented, which resulted in a smaller number of packets to create and manage the DODAG and improve the performance of the proposed method. Thereupon, using the idea of a multi-class RPL with virtualization can play a significant role in the smart grid and can be utilized as an alternative solution.

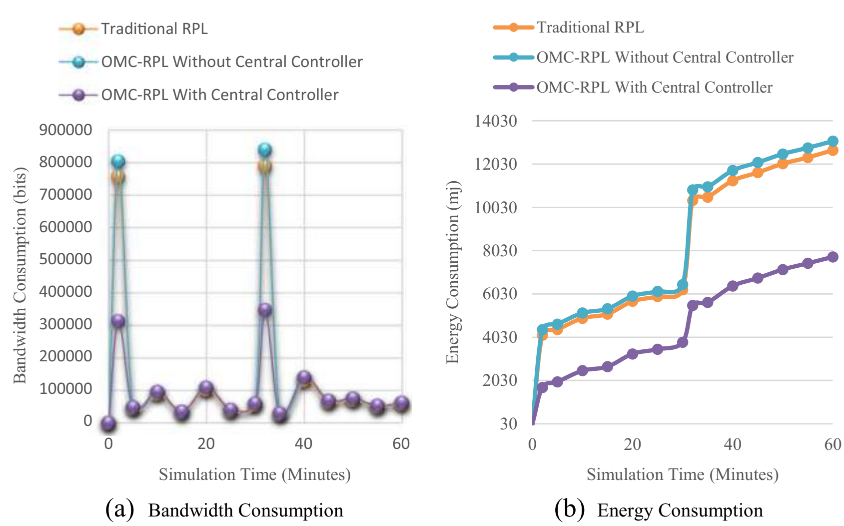


Fig. 14 Comparison of the bandwidth and energy consumption

References

1. Gungor VC, Sahin D, Kocak T, Ergut S, Buccella C, Cecati C, Hancke GP (2013) A survey on smart grid potential applications and communication requirements. IEEE Trans Ind Inf 9(1):28–42. doi:10.1109/TII.2012.2218253
2. Ho Q-D, Gao Y, Le-Ngoc T (2013) Challenges and research opportunities in wireless communication networks for smart grid. IEEE Wirel Commun 20(3):89–95
3. Rajalingham G, Gao Y, Ho Q-D, Le-Ngoc T (2014) Quality of service differentiation for smart grid neighbor area networks through multiple RPL instances. In: Proceedings of the 10th ACM symposium on QoS and security for wireless and mobile networks. ACM, p 17–24
4. Thai P (2011) Dissertation/thesis approval form
5. Winter T (2012) RPL: IPv6 routing protocol for low-power and lossy networks
6. Vasseur J (2014) Terms used in routing for low-power and lossy networks. In
7. Gaddour O, Koubâa A (2012) RPL in a nutshell: a survey. Comput Netw 56(14):3163–3178
8. Thubert P (2012) Objective function zero for the routing protocol for low-power and lossy networks (RPL)
9. Gnawali O, Levis P (2010) The ETX objective function for RPL
10. Vasseur J-P, Kim M, Pister K, Dejean N, Barthel D (2012) Routing metrics used for path calculation in low-power and lossy networks. In
11. Gaddour O, Koubaa A, Chaudhry S, Tezeghdanti M, Chaari R, Abid M (2012) Simulation and performance evaluation of DAG construction with RPL. In: Communications and Networking (ComNet), 2012 Third International Conference on. IEEE, p 1–8
12. Wang D, Tao Z, Zhang J, Abouzeid A (2010) RPL based routing for advanced metering infrastructure in smart grid. In: Communications Workshops (ICC), 2010 I.E. International Conference on. IEEE, p 1–6
13. Long NT, Uwase M-P, Tiberghien J, Steenhaut K (2013) QoS-aware cross-layer mechanism for multiple instances RPL. In: Advanced Technologies for Communications (ATC), 2013 International Conference on. IEEE, p 44–49
14. Abdessalem RB, Tabbane N (2014) RPL-SCSP: a network-MAC cross-layer design for wireless sensor networks. In: Proceedings of Ninth International Conference on Wireless Communication and Sensor Networks. Springer, p 27–35
15. Zhang J, Seet B-C, Lie T-T, Foh CH (2013) Opportunities for software-defined networking in smart grid. In: Information, Communications and Signal Processing (ICICS) 2013 9th International Conference on. IEEE, p 1–5
16. Islam MM, Hassan MM, Lee G-W, Huh E-N (2012) A survey on virtualization of wireless sensor networks. Sensors 12(2):2175–2207
17. Tan K-L What's NExT?: Sensor + Cloud!? Paper presented at the Proceedings of the Seventh International Workshop on Data Management for Sensor Networks, Singapore
18. Rutakemwa MM (2013) From physical to virtual wireless sensor networks using cloud computing. Int J Res Comput Sci 3(1):19
19. Misra S, Chatterjee S, Obaidat MS (2014) On theoretical modeling of sensor cloud: a paradigm shift from wireless sensor network
20. Kong PY (2015) Wireless neighborhood area networks with QoS support for demand response in smart grid. Smart Grid, IEEE Transactions on PP(99):1–1. doi:10.1109/TSG.2015.2421991
21. Deshpande JG, Kim E, Thottan M (2011) Differentiated services QoS in smart grid communication networks. Bell Labs Tech J 16(3):61–81
22. Gharavi H, Xu C (2012) Traffic scheduling technique for smart grid advanced metering applications. IEEE Trans Commun 60(6):1646–1658
23. Ramírez DF, Céspedes S (2015) Routing in neighborhood area networks: a survey in the context of AMI communications. J Netw Comput Appl 55:68–80
24. Di Marco P, Fischione C, Athanasiou G, Mekikis P-V (2013) MAC-aware routing metrics for low power and lossy networks. In: INFOCOM, 2013 Proceedings IEEE. IEEE, p 13–14
25. Brachman A (2013) Rpl objective function impact on llns topology and performance. In: Internet of things, smart spaces, and next generation networking. Springer, p 340–351
26. Karkazis P, Leligou HC, Sarakis L, Zahariadis T, Trakadas P, Velivassaki TH, Capsalis C (2012) Design of primary and composite routing metrics for rpl-compliant wireless sensor networks. In: Telecommunications and Multimedia (TEMU), 2012 International Conference on. IEEE, p 13–18
27. Gaddour O, Koubâa A, Baccour N, Abid M (2014) OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol. In: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on. IEEE, p 365–372
28. Tripathi J, De Oliveira JC, Vasseur J-P (2014) Proactive versus reactive routing in low power and lossy networks: performance analysis and scalability improvements. Ad Hoc Netw 23:121–144
29. Gaddour O, Koubâa A, Abid M (2015) Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL. Ad Hoc Netw 33:233–256
30. Barriquello CH, Denardin GW, Campos A (2015) A geographic routing approach for IPv6 in large-scale low-power and lossy networks. Comput Electr Eng 45:182–191
31. Ancillotti E, Bruno R, Conti M (2014) Reliable data delivery with the ietf routing protocol for low-power and lossy networks. IEEE Trans Ind Inf 10(3):1864–1877
32. Dorsch N, Kurtz F, Georg H, Hagerling C, Wietfeld C (2014) Software-defined networking for Smart Grid communications: Applications, challenges and advantages. In: Smart Grid Communications (SmartGridComm), 2014 I.E. International Conference on. IEEE, p 422–427
33. Sydney A, Ochs DS, Scoglio C, Gruenbacher D, Miller R (2014) Using GENI for experimental evaluation of software defined networking in smart grids. Comput Netw 63:5–16
34. Molina E, Jacob E, Matias J, Moreira N, Astarloa A (2015) Using software defined networking to manage and control IEC 61850-based systems. Comput Electr Eng 43:142–154
35. Su Z, Xu Q, Zhu H, Wang Y (2015) A novel design for content delivery over software defined mobile social networks. IEEE Netw 29(4):62–67
36. Tuna G, Gungor VC, Gulez K (2013) Wireless sensor networks for smart grid applications: a case study on link reliability and node lifetime evaluations in power distribution systems. Int J Distrib Sens Netw 2013
37. Fadel E, Gungor V, Nassef L, Akkari N, Maik MA, Almasri S, Akyildiz IF (2015) A survey on wireless sensor networks for smart grid. Comput Commun 71:22–33
38. Gopi C, Lalu V (2016) Sensor network infrastructure for AMI in smart grid. Procedia Technol 24:854–863
39. Salvadori F, Gehrke CS, de Oliveira AC, de Campos M, Sausen PS (2013) Smart grid infrastructure using a hybrid network architecture. In: IEEE Transactions on Smart Grid, vol. 4, no. 3, p 1630–1639
40. Al-Anbagi I, Erol-Kantarci M, Mouftah HT (2014) Priority- and delay-aware medium access for wireless sensor networks in the smart grid. In: IEEE Systems Journal, vol. 8, no. 2, p 608–618
41. Shah S, Thubert P (2012) Differentiated service class recommendations for LLN traffic
42. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Neural networks, 1995. Proceedings., IEEE International Conference on, Nov/Dec 1995, vol. 1944. p 1942–1948
43. Rini DP, Shamsuddin SM, Yuhaniz SS (2011) Particle swarm optimization: technique, system and challenges. Int J Comput Appl 14(1):19–26
44. Engelbrecht AP (2006) Fundamentals of computational swarm intelligence. Wiley

Mohammad Alishahi received his B.S. and M.S. degree in computer engineering from the Islamic Azad University of Mashhad, Iran in 2010. He is currently Ph.D student at the Islamic Azad University, Science and Research Branch, Tehran, Iran. He was also a research visitor at UCLA Smart Grid Energy Research Center. His research interests include power and communication network analysis, applications of smart energy systems and routing.

Hamidreza Pourreza received his B.S. degree in Electrical Engineering from Ferdowsi University of Mashhad, 1989, and M.S. degree in Electrical Engineering from Tehran Polytechnic (Amirkabir) University of Technology in 1993. He received his Ph.D degree in Computer Engineering from Tehran Polytechnic (Amirkabir) University of Technology in 2003. He is an Associate Professor at Computer Engineering Department, Ferdowsi University of Mashhad. His research interests are in Computer Vision, Signal Processing, Hardware Design and Intelligent Transportation Systems (ITS).

Mohammad Hossein Yaghmaee Moghaddam was born in July 1971 in Mashhad, Iran. He received his B.S. degree in Communication Engineering from Sharif University of Technology, Tehran, Iran in 1993, and M.S. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 1995. He received his Ph.D degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 2000. His research interests are in Smart Grid, Computer and Communication Networks, Quality of Services (QoS), Wireless Sensor Networks (WSN) and fuzzy logic control was with Department of Electrical and Computer Engineering, University of Toronto (UoT), as a visiting professor.